

Available online at www.sciencedirect.com**ScienceDirect**

Procedia Computer Science 32 (2014) 317 – 324

Procedia
Computer Science

5th International Conference on Ambient Systems, Networks and Technologies (ANT-2014)

Mobile agent based train control system for mitigating meet conflict at turnout

Anshul Verma*, K. K. Pattanaik

*Information and Communication Technology**Atal Bihari Vajpayee-Indian Institute of Information Technology and Management, Gwalior, India.*

Abstract

Timeliness and correctness in conflict detection and resolution mechanisms form an important component in railway operation especially when the decision process involves human factors. The current approaches used by Indian railways (IR) for conflict management can be enhanced through the use of Mobile Agent (MA) technology. The use of MA technology appropriates due to their asynchronous, autonomous, robust, and fault tolerant properties. In this paper, we discuss certain train operation conflicts and detail about *meet conflict* followed by implementation of MA based *meet conflict* management system for IR. Due to the suitability of formal modeling techniques for implementing highly dynamic and concurrent systems, in our rail traffic scenario we modeled the *meet conflict* detection and resolution mechanism using coloured petri net (CPN).

© 2014 The Authors. Published by Elsevier B.V. Open access under [CC BY-NC-ND license](#).
Selection and Peer-review under responsibility of the Program Chairs.

Keywords: Indian railways; Mobile agent; Railway traffic; Conflict detection; CPN.

1. Introduction

The traditional approach of axle count and induction loop based method to monitor the rail track occupancy involves control and operation by drivers based on trackside interlocking and blocking, in conjunction with various train signaling and surveillance devices, as well as human experience and skills. While conflict scenarios are taken care of while designing the train running diagrams. In run time scenarios, when the trains fall out of schedule, the respective station controllers re-schedule the affected trains in the unoccupied track sections based on a coordinated

* Corresponding author. Tel.: +91-9826074618.
E-mail address: anshulverma87@gmail.com

decision among respective station controllers of the adjoining stations. Manual approach is followed in IR to deal with such scenarios. For uniform traffic scenarios such approach works well, however in mixed traffic scenarios such approach at times can cause conflicts in train operations. Trains are operated in IR according to General rules of IR [1]. Computer based conflict management is not a new approach in many countries where the operation is based on Communication Based Train Control (CBTC) system [2] and unlike the mixed traffic in IR the traffic type is mostly uniform. The infrastructural setup for CBTC system operates on moving block signaling system [3]. Whereas IR operates on static block signaling system [4] and the infrastructure for both the signaling systems are different. This in addition to mixed traffic scenario of IR causes the mechanisms meant for moving block signaling system cannot be directly applied on an existing static block signaling system. This necessitates careful investigation of the existing system of IR in order to enhance the system from its present level to CBTC based system. MA based systems can be a design choice for CBTC system due to their asynchronous, autonomous, robust, and fault tolerant properties.

The safe (conflict free) operation of rail system depends essentially on the performance of its train control system composed of multi-section with interconnecting and interaction mechanism. The train control system uses run time information and other environmental data in combination with onboard references for taking train rescheduling decisions when the trains fall out of the reference train running diagram. Rescheduling introduces conflict situations with respect to reference train diagram and are handled by station controllers manually.

Run time automatic conflict management is one of the important areas of research in train traffic operation. In our work we focused specifically on the run time automatic conflict management using MA approach in railway operations adjustment arising due to rescheduling decisions. The model can be used to impress upon the reliability of such automated systems in train operation where a high degree of precision is augmented. However due to space limitations in this paper we restrict our discussion to the MA based run time automatic *meet conflict* detection and resolution mechanism.

The rest of the paper is organized as follow. Section 2 covers the related work and Section 3 describes the system. Conflict modeling and implementation is covered in Section 4 and result analysis is presented in Section 5. Finally Section 6 concludes the paper.

2. Related work

Several approaches based on mathematical models, simulations, mobile agents, and soft computing techniques corresponding to static block signaling and moving block signaling systems have been dealt in the literature. We confine our discussion to the recent efforts on autonomous vehicle control system research with emphasis on automatic conflict management.

A realistic simulation environment using CPN for working on Vehicle-on-board Automatic Train Protection (V-ATP) is provided [5] as a safety critical subsystem. The hierarchy based system relied on balise, the track side electronic devices for communication with the train. In addition to calculating safety parameters corresponding to each train, their work emphasized more on the how to refine the basic CPN based model for further research.

The train conflicts and classification of them into 7 categories is given in [6]. The prediction based train operation conflict detection method is proposed while considering various uncertainties of train operation. The paper did not report any outcome in terms of conflict resolution management.

A new approach for the implementation of a completely autonomous multi-agents advanced FPGA based train control system is described [7]. The proposed system supports several issues related to stability and controllability, provides relevant safety and operational requirements by using communication methods and various sensors.

An expert system approach applied to train operation adjustment under moving block signaling system is presented in [8]. The approach of dynamic construction of train running diagram for operation adjustment was followed.

MARCS - Multi-Agent Railway Control System [9] is a multi-layered system comprising three distinct agent types, i.e., Supervisor, Train and Station. The Learning layer creates rules by learning from the situations accumulated by the Control layer, these rules can improve traffic control processes, minimizing waiting time and stop orders sent for each train.

A hybrid software agent model for designing Chinese Train Control System (CTCS) simulation software on the basis of deliberative and reactive agent concepts is developed [10]. The reactive memory used in the model appropriates the agents' use in real-time. The authors have concluded the efficacy of the application of their hybrid software agent to distributed supervising and control systems.

A MA based Freight Trains Traffic Management System using CaseLP, a logic programming based environment for MA system prototyping, is presented [11]. The advantages have been successfully demonstrated on the Italian railway line between Milano and La Spezia stations.

The TeleTruck approach, which is a MA system with holonic agents to support transport scheduling in real-world situations like dynamics and uncertainty is described [12]. They also discussed requirements of holonic system in terms of communication, coordination, and resource control, and implemented these functionalities in their prototype to make the dispatch support system better.

A MA system for planning and plan execution monitoring in a dynamic railway transportation environment is presented [13]. The planning process includes incremental anytime algorithm to integrating new task specifications in the ongoing planning process. The plan execution monitor is used to run-time monitor ongoing driving or coupling activities of railway transportation.

A MA system to solve vehicle-routing problem in a highly dynamic and unpredictable environment by applying an in-time algorithm that is capable to quickly find out best solutions to a problem is described [14]. They used a stochastic improvement mechanism to achieve high quality solutions in a distributed environment.

A MA based CBTC framework with moving block signaling using MESSAGE methodology for IR is presented [15]. It was a work-in-progress paper and a simplified model of sub-goal moving authority given was implemented using CPN to evaluate the behavioral aspects. It did not discuss in detail the agents' reactive behavior for various conflict scenarios and their cooperative interaction for resolving them.

To the best of our understanding none of the work focused specifically on the run time automatic conflict management using MA approach in railway operations adjustment arising due to rescheduling decisions.

3. System description

Conflicting scenarios arise at many stages of railway operation as the infrastructure is shared among running trains. Identification, modeling, and testing of various conflict situations along with their resolution mechanisms would serve as a first step towards building a robust system. Moreover, the system being highly dynamic and concurrent in nature the test of the system majorly depends on the efficiency of these basic building blocks. The major conflict scenarios mentioned in [6] are *Meet conflict*, *Pass conflict*, *Safety intervals at the station*, and *Capacity conflict*. Train operation prediction approach for avoiding conflict is a technique used while preparing train running diagram. However, train operation prediction during run time scenario may be quite different due to various reasons and cause the trains to fall out of schedule. Such out of schedule trains may create a conflict scenario. In such cases conflict detection and on-the-fly resolution mechanisms are required. We first discuss the *meet conflict* scenario followed by its modeling. Modeling terminologies used in our work are briefly described below.

IR operates on static block signaling and uses electronic interlocking method for *meet conflict* management. Turn out on a railway track is a track switching location enabling trains to be guided from one railway track to another. Interlocking is a process by which route is established for the trains through a combination of setting of turnout switches and aspect signaling. *Meet conflict* is a situation at turnout where two inbound trains approach the turnout simultaneously and may collide at turnout if sufficient care is not taken to schedule those trains. The proposed CPN model of the system is based on the following three major components to describe the *meet conflict* scenario and its resolution.

3.1. Permanent way infrastructure

- *Train*: All tokens of type *TRAIN2* represent trains and the token attributes represent control information carried by the train.
- *Track*: Track is modeled as a collection of continuous places and arc represents the direction of movement. For a conflict free operation it is desired for a place to have at most one train.

- *Track side devices*: Active RFID devices and Laser speed guns used to capture train information and precision speed respectively. These devices are installed at strategic locations along the track. These information are carried by MA to be processed at MA collaborative point for meet conflict resolution.

3.2. Operation

- *Train Movement*: Movement of train from one discrete point to other is treated as an event that is an outcome of firing of corresponding transition. The journey of train can be considered as a collection of several such events.
- *Dispatch Time*: Time stamp associated with token of type *TRAIN2* represents dispatch time of the corresponding train at that place.
- *Workload*: The model comprises two input flows of trains approaching the turnout.
- *Output flow*: Output flow comprises trains passing through turnout.
- *Traversal time*: Traversal time describes the time a train takes to travel between two places. A time region is added (starting with @+) to all transitions to model the traversal time.

3.3. Control mechanism

- *Mobile Agent*: MA functionalities are put into transition. When a transition executes it consumes token(s) and also produce token(s). This signifies that the MA accepting/generating input from/to other MA. While an arc represents the direction of flow of information. Agents constitute the main control infrastructure for the train operation to take place. Agents carry information in run time, collaborate at suitable points of computation to decide on train movement from one place to another on the track, and deliver it to train.
- *Conflict free operation*: It is a direct consequence of functionalities of MAs.

4. System modeling

This section describes the major situations like rescheduling of unscheduled trains and conflict management. Finally, we describe the system implementation in CPN.

4.1. Major situations

- *Rescheduling*: We categorize the trains in to two broad categories on the basis of their current running schedule: "scheduled", and "unscheduled". The unscheduled trains' arrivals are modeled as Poisson process. Hence the total workload consist of mix of scheduled (at places *pin1* and *pin2*) and unscheduled (at places *pd1* and *pd2*) trains (see Fig. 3 and 4). The speed variation of a train during its course of run is modeled as simple random process and is implemented in transitions *tin1* and *tin2*.
- *Conflict management*: To avoid conflict situation arising at turnouts the conflict detection and resolution mechanism is modeled. The MA based infrastructure is the enabler for the conflict management system. MA functionalities are put into transitions. The agents are used to decide firing of the transition based on the interaction mechanism which are based on the firing rules.

The track conflict detection is modeled by recording the succession track occupancy status with the help of single token constraint at places *la1*, *la2*, *lb1*, *lb2*, and *lc1*. Presence of a token at the above places represents the corresponding track section is empty. The detection for any possible *meet conflict* of the approaching trains at the turnout is decided at transitions *ta2* and *tb2* based on the trains' priorities and speeds, and a proactive decision is taken to avoid conflict at turnout by adjusting their speeds. Speed adjustment is modeled by introducing delay in to the operation.

4.2. System implementation

The functions used in the model are shown as snapshot in Fig. 1. Initializations and assumptions (if any) are described as required while explaining a function.

The function $TSS(s)$ describes a time region of all places except $ta2$ and $tb2$. Each train has a time stamp, which represents departure time of the corresponding train. This time region $@++TSS(s)$ represents, the time, a train takes during movement between consecutive places. In our model the distance between consecutive places is taken as 200 meters. Initially the values for the speed parameter are generated randomly between 25 to 35 distance units. This is to model the speed variations in train operation due to environmental uncertainties. The speed is assumed to remain constant for the rest part of the simulation.

The functions $camp1(pr2, s, p1, s1)$ and $camp2(pr2, s, p1, s1)$ are the functionalities of Agents used to proactively check for any conflicting scenarios at the turnout. If detected, the trains' speeds are adjusted while maintained a safety factor. Function $RDEL()$ is used as time region at transitions $td1$ and $td2$ to introduce random delays to the trains. This is done to model the scenario of environmental uncertainties leading to delays in train operation. Function $RS()$ is used to randomly update the speed of trains at transitions $tin1$ and $tin2$. This is to model the scenario of speed variations due to environmental uncertainties.

Most abstract page top: The most Abstract description of the hierarchical version of *meet conflict* model is shown in Fig. 2. It tells us that we have two incoming tracks, one outgoing track and one turnout with *meet conflict* detection and resolution logic. The box next to the substitution transition specifies the subpage that contains the detailed description of the activity. Each subpage has a number of port places, marked with an In-tag or Out-tag. They constitute the interface through which the subpage communicates with its surroundings.

Subpage inflow1: This subpage represents an incoming track to a turnout (see Fig. 3). We are taking a small track for our convenience. We can replace this with any real life incoming railway track. We relate the input port $la1$ in Fig. 3 to the input socket $la1$ in Fig. 2 and output port $A1$ in Fig. 3 to the output socket $A1$ in Fig. 2. It consists of a single track segment from place $pin1$ to place $A1$. We introduce a single place $la1$, to limit the number of trains i.e., only a single train can be stayed at a time on places $A1$. The subpage also contains rescheduling and train movement model components as described earlier.

Subpage inflow2: Analogously, This subpage also represents an incoming track to a turnout (see Fig. 4). The designing components and functionality of them are same as in subpage *inflow1*.

Subpage meet: The subpage describes the logic of *meet conflict* detection and resolution at turnout as shown in Fig. 5. The subpage *meet* has six port places, each has the same name as the socket to which it is assigned as shown in Fig. 2.

The subpage consists of three track segments, segment1 from places $A1$ to *turnout*, segment2 from $A2$ to *turnout*, and segment3 from *turnout* to $A3$. We introduce six places $la1$, $la2$, $lb1$, $lb2$, $lc1$ and $lc2$, to limit the number of trains on the track. The places $ppr1$ and $ppr2$ are used to transmit priority and speed values of trains for comparison purpose. When trains pass transition $ta1$ and $tb1$, the initial token $(0, "0.0")$ of corresponding places $ppr1$ and $ppr2$ are updated with passed train's priority and speed values. In the next step, token values of places $ppr1$ and $ppr2$ are used for comparison purpose during the execution of corresponding transitions $tb2$ and $ta2$. When trains pass transition $ta3$ and $tb3$, the token values of corresponding places $ppr1$ and $ppr2$ are again initialized by $(0, "0.0")$.

As described earlier under the heading model component, Conflict management is the part of this subpage. we compare the priorities of two trains of different track segments, which may conflict at the turnout, to detect the conflict. If conflict is detected, we delay the lower priority train by adding a greater time region value and dispatch the higher priority train earlier by adding normal time region value to resolve the conflict.

Subpage outflow1: This subpage represents an outgoing track from a turnout as shown in Fig. 6. Trains of both incoming tracks pass through this outgoing track. Places $A3$ and $lc2$ are used as connector between subpages *outflow1* and *meet*.

The diagram illustrates a causal model with the following components and relationships:

- Nodes:**
 - inflow1** (grey rectangle)
 - inflow2** (blue rectangle)
 - A1** (white circle)
 - A2** (blue circle)
 - lb1** (white circle)
 - lb2** (white circle)
 - meet** (red rectangle)
 - A3** (white circle)
 - outflow1** (red rectangle)
- Edges and Labels:**
 - inflow1** to **A1**: Black arrow labeled **TRAIN2**.
 - inflow2** to **A2**: Blue arrow labeled **TRAIN2**.
 - lb1** to **lb2**: Red arrow labeled **E**.
 - lb2** to **meet**: Red arrow labeled **1^e**.
 - lb1** to **meet**: Red arrow labeled **E**.
 - A1** to **meet**: Black arrow.
 - A2** to **meet**: Blue arrow labeled **meet₁**.
 - meet** to **A3**: Red arrow labeled **TRAIN2**.
 - A3** to **outflow1**: Red arrow labeled **TRAIN2**.
 - lb2** to **A3**: Red arrow labeled **1^e**.

Fig. 2. Most abstract page *Top*.

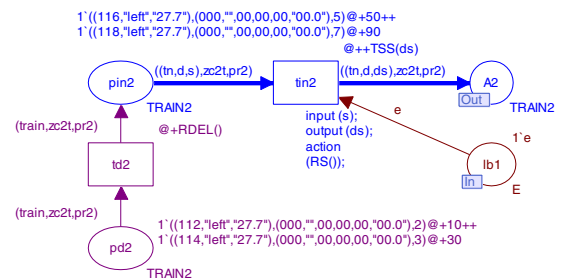


Fig. 4. Subpage *inflow2*.

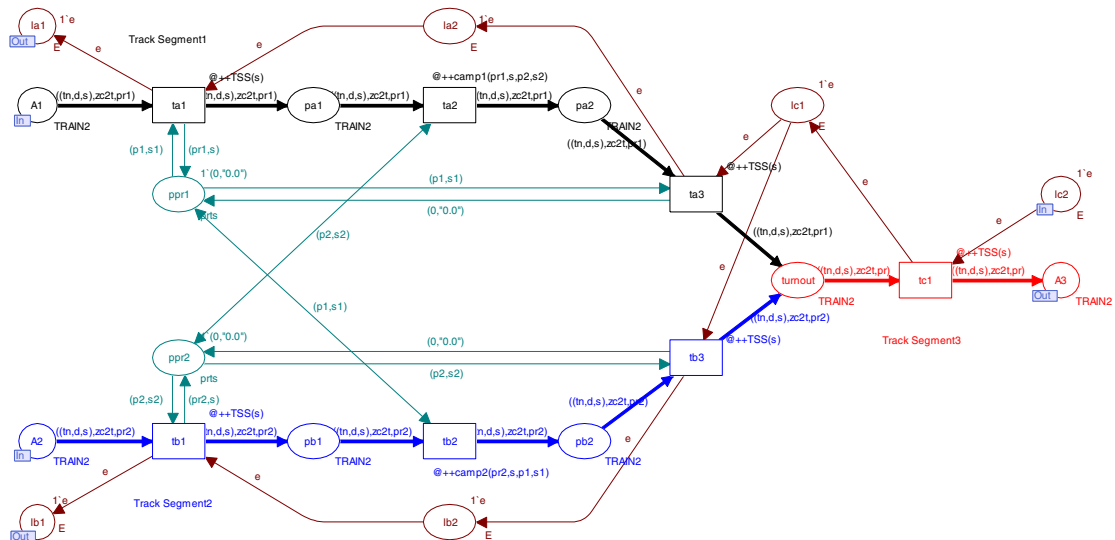


Fig. 5. Subpage *meet*.

The behavioral analysis of the CPN model representing the system is required to check the correctness of the model. CPN provides two different analysis mechanisms to validate the CPN model: Simulation Report and State Space Analysis [16][17].

To construct the Simulation and State Space Report of reasonable size, and to generate various conflict situations we considered one dispatch schedule at a time from the many possible schedules. Due to page restrictions we present only the summary of the state space analysis.

5.1. Simulation report analysis

Simulation report contains detailed information about all the occurred binding elements. The readings pertaining to conflict situations are shown in Table 1. The first, second and third parameters represent step number, model time and transition name respectively. The information after the @ sign specifies the page to which the occurring transition belongs. The information after the - sign describes incoming arc variable and its value.

In the simulation report, the steps 5 and 6 represent that during the execution of transition *ta2* and *tb2* at model time 24.44, each transition has a train which are going to conflict at turnout, because the distance of turnout from these transitions are same, and both trains are also running with same speed. This situation represents a *meet conflict* at turnout. The model detects this situation and solves by adding an adequate delay to the train with lower priority. Steps 7 and 8 represent that only higher priority train 112 is allowed to travel through turnout by firing transitions *tb3* and *tc1* at model time 31.66 and 38.88 respectively. Thereafter, step 9 represents that the lower priority train 111 is allowed to travel by firing transitions *ta3* at model time 46.10, when the successor train has passed the turnout.

Similarly, the conflict situations are detected and resolved between rest two pairs of conflicting trains with train number 113-114 and 115-116. The remaining trains with train number 117 and 118 have different running schedule, therefore, conflict does not occur. Thus, the simulation report proves that the model is behaving as required.

5.2. State space analysis

State space is a graph containing a node for each reachable marking and an arc for each occurring binding element. Simulation report is very useful for the verification and understanding of CPN model step by step, but it does not provide complete proof of dynamic properties of the model. We analyze the State space report to study the reachability, boundness, reversibility and home state, liveness, and fairness properties.

The first part contains statistical information about the size of the state space as shown in Table 2. There are equal numbers of strongly connected components (SSC) and state space nodes, means it has no SSC sub-graphs with more than one node. It represents that model does not have any cycle, means model must terminate after some finite number of steps as was intended.

Table 1. Some readings of simulation report.

5 24.44 tb2 @ (1:meet)	7 31.66 tb3 @ (1:meet)
- d = "left"	- s2 = "27.7"
- pr2 = 2	- p2 = 2
- s = "27.7"	- d = "left"
- zc2t = (0,"",0,0,0,"00.0")	- pr2 = 2
- tn = 112	- s = "27.7"
- s1 = "27.7"	- zc2t = (0,"",0,0,0,"00.0")
- p1 = 1	- tn = 112
6 24.44 ta2 @ (1:meet)	8 38.88 tc1 @ (1:meet)
- d = "left"	- d = "left"
- s = "27.7"	- s = "27.7"
- zc2t = (0,"",0,0,0,"00.0")	- zc2t = (0,"",0,0,0,"00.0")
- pr1 = 1	- tn = 112
- tn = 111	- pr = 2
- s2 = "27.7"	9 46.10 ta3 @ (1:meet)
- p2 = 2	- s1 = "27.7"
	- p1 = 1
	- d = "left"
	- s = "27.7"
	- zc2t = (0,"",0,0,0,"00.0")
	- pr1 = 1
	- tn = 111

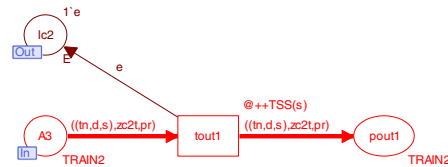


Fig. 6. Subpage *outflow1*.

Table 2. Size of state space and SCC graph.

Statistics
State Space
Nodes: 59
Arcs: 68
Secs: 0
Status: Full
Scc Graph
Nodes: 59
Arcs: 68
Secs: 0

6. Conclusions

The structure of conflict detection scenario in rail network has been modeled using CPN. The conflict detection in the moving trains by MAs at the turnout has been tested for multiple flows of trains. The model incorporates few run time situations that arise during trains' running in order to simulate the actual train running scenario. However, a lot of other factors influencing the train operation need to be considered for incorporation to our basic model. The behavioral properties of the CPN model have been studied extensively in terms of reachability, boundedness, liveness, and fairness from the system generated report. Dynamic estimation of line headway for the same scenario would be taken up next.

Acknowledgements

This report is an interim exploratory result of the project named "Multi Mobile Agent based Train Operation in Moving Block Setup" which is sponsored by Department of Information Technology, Ministry of Communications and Information Technology, Government of India.

References

1. South Central Railway, General rules for Indian Railways, 2008.
2. Yong, Zhang. Ming, Zhao. Xi-shi, Wang. "Research on a Train Operation Simulation System under Moving Automatic Block Conditions," *Journal of system simulation*, VOL. 11, NO. 3, 1999
3. Zhi, Zhao. Yin-hang, Cheng, "The comparison of several moving block systems," *Railway signal and communication*, VOL. 35, NO. 12, 1999.
4. Goundan, P.R., Jhunjhunwala, A., "Axle counter based block signalling for safe and efficient train operations," *Vehicular Technology Conference*, 1999. VTC 1999 - Fall. IEEE VTS 50th, vol.2, no., pp.824-828 vol.2, 1999.
5. Lei Chen; Bin Ning, Tian-Hua Xu, "Research on modeling and simulation of vehicle-on-board Automatic Train Protection subsystem of Communication Based Train Control system," *IEEE International Conference on Vehicular Electronics and Safety*, pp. 1-5, 2007.
6. Wen Chao, "Prediction of train operation conflict," *Control Conference (CCC)*, 29th Chinese, pp. 5441 - 5446, 2010.
7. M. Saleem Khan, Khaled Benkrid, "Development Techniques of Multi-Agents Based Autonomous Railway Vehicles Control Systems," *World Academy of Science, Engineering and Technology* 72, pp. 558-568, 2009.
8. Zhang Li-yan, Li Ping, Jia Li-min, Yang Feng-yan, "Study on the simulation for train operation adjustment under moving block," *Intelligent Transportation Systems*, pp. 351 - 356, 2005.
9. Hugo Proenca and Eugenio Oliveira, "MARCS Multi-Agent Railway Control System," published in *Lecture Notes in Computer Science, Advances in Artificial Intelligence - IBERAMIA 9th Ibero-American Conference on AI*, Puebla, Mexico, pp. 1-10, Nov. 2004.
10. Xiao-Hui Hu, Xing-She Zhou, Jian-Wu Dang, "A Designing Method of Simulation Software for Chinese Train Control System Based on Hybrid Software Agent Model," *International Conference on Machine Learning and Cybernetics*, pp. 148-153, 2006.
11. Cuppari, P. L. Guida, M. Martelli, V. Mascardi, and F. Zini, "Prototyping freight trains traffic management using multi-agent systems," *International Conference on Inf. Intell. Syst.*, Los Alamitos, CA, pp. 646-653, 1999.
12. H.-J. Burckert, K. Fischer, and G. Vierke, "Transportation scheduling with holonicMAS: The TELETRUCK approach," *3rd International Conference on Practical Appl. Intell. Agents Multi-Agent Technol.*, Blackpool, U.K., pp. 577-590, 1998.
13. J. Lind, K. Fischer, J. Bocker, and B. Zirkler, "Transportation scheduling and simulation in a railroad scenario: A multi-agent approach," *4th International Conference on Practical Appl. Intell. Agents Multi-Agent Technol.*, Blackpool, U.K., pp. 325-344, 1999.
14. R. Kohout and K. Erol, "In-time agent-based vehicle routing with a stochastic improvement heuristic," *11th Conference on Innovative Applications of Artificial Intelligence*, Menlo Park, CA, pp. 864-869, 1999.
15. Anshul Verma, K.K. Pattanaik, P.P. Goel, "Mobile Agent based CBTC System with Moving Block Signalling for Indian Railways," *2nd International Conference on Railway Technology: Research, Development and Maintenance*, France, July 2014, unpublished.
16. K. Jensen. "Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use. Volume 1: Basic Concepts." Springer-Verlag, 1992.
17. K. Jensen, "Coloured Petri Nets. Basic Concepts, Analysis Methods and Practical Use. Volume 2: Analysis Methods", Springer-Verlag, 1994.